Assignment 1 (Individual Assignment)

Due Date

19 Mar 2022 17:00 (Saturday)

Declaration of Original Work

Plagiarism is a serious misconduct. No part of students' assignment should be taken from other people's work without giving them credit. All references must be clearly cited. Any plagiarism found in students' completed assignments can lead to disciplinary actions such as mark deduction, disqualification or even expulsion by the College.

By submitting this assignment to the subject lecturer through Moodle, you hereby declare that the work in this assignment is **completely** your own work. No part of this assignment is taken from other people's work without giving them credit. All references have been clearly cited.

You understand that an infringement of this declaration leaves you subject to disciplinary actions such as mark deduction, disqualification or even expulsion by the College.

Instruction

Each of you is required to submit a MS Word 2016 file (**with the template provided**) showing your answers, programs and testing screen dumps. Your MS Word file should contain a cover page showing your name, your student ID no, course code (SEHH2239), course name (Data Structures), class (201) and date.

All programs must be written in Python programming language. For each programming-type question, please also write down the above-mentioned particulars as comments on the .py source file, compile and test it with at least three testing cases. Show the program source code and the screen dumps of testing in the MS Word file. Each program must be successfully compiled and can execute, otherwise 50% of the total marks of that question will be deducted.

All submitted assessments will be evaluated with Python version 3.7.12 (the current python version used in Google Colab). Your submitted assessments must run without errors on Google Colab.

Unless otherwise instructed, you **MUST NOT** import any modules in your submitted assessments

Items to be Submitted

1. **For each student,** you need to convert the MS Word to **PDF** file before submission. Submit a softcopy to Moodle

2. **For each student,** put the MS Word file together with the program files (both .py source files) into a folder (**folder name format:** Student Name_Student ID No e.g. ChanTaiMan_16001234A). Use a compression software (e.g. Winzip) to compress the folder (e.g. ChanTaiMan_16001234A.zip) (***Submit the compressed folder e.g. ChanTaiMan_08001234A.zip via individual class Moodle. In case the Moodle's Assignment Submission System is not available, send the softcopy to me through e-mail***)

Attention:

While submitting the **softcopies** via Moodle, a timestamp will be placed on the softcopies of your assignment. There will be a sharp cut-off time at Moodle, so late assignments will be recorded at Moodle. Softcopies submitted via email or other means will NOT be accepted unless the Moodle is not available. As many students will submit their assignments to Moodle at around the deadline time, it normally takes longer for uploading your assignment, so it is strongly suggested that you start submitting earlier, say at least 45 minutes before the deadline. Marks will be deducted for late submission. Successful submission of this assignment includes the submission of both softcopy and hardcopy. Missing either softcopy or hardcopy is not successful submission.

Plagiarism will be penalized severely. Marks will be deducted for assignments that are plagiarized in whole or in part, regardless of the sources.

Late submission is liable to a **penalty of 10%** of the available marks for **each day late**; Saturdays, Sundays and holidays are counted.

Question 1

Write a Python program that

(a) With a Python class called ***Employee*** with the following features:

1. Two instance variables
   name, data type: String
   salary, data type: integer

2. One constructor with two parameters
   initialize the member variable *name*
   initialize the member variable *salary*

(b) Create a list **EmpList**

(c) Put the following employee information in the list created by step (b)

name	**salary**
Ada	15000
Brian	18000
Carson	12000
Dave	14000

Question 2

Refer to the question 1, add the following in ***Employee*** class

(a) A method `insertionSort` to accept Employee list and sort the list by using their salary

(b) Call `insertionSort` method to sort the list by using employee's salary

(c) Print the information of each ***Employee*** before and after sorting (refer to the following sample output)

Sample output

```
Before Insertion Sort:
Ada,15000
Brian,18000
Carson,12000
Dave,14000

After Insertion Sort:
Carson,12000
Dave,14000
Ada,15000
Brian,18000
```

Please provide 3 more testing cases.

Question 3

(a) State the differences between bubble sort and quick sort. Use your own wordings.

(Hint: You can compare these 2 sorting algorithms by algorithm description, algorithm efficiency, implementation, etc.)

(b) Choose a sorting algorithm that is based on **recursion**, provide the **pseudo code**, and describe how it use recursion to achieve sorting. Use your own wordings.

Question 4

You should form the list of integers with your student ID. The numbers are generated as follows:

Number	Based on your student ID number, formed by its …
a	1st and 2nd digits
b	2nd and 3rd digits
c	3rd and 4th digits
d	4th and 5th digits
e	5th and 6th digits
f	6th and 7th digits

For example, the student ID is 20563755A.
Values of *a* to *g* are: 20, 5, 56, 63, 37, 75
(If the starting digit is 0, just remove it)

Sort them using the following algorithms and **ALL** the steps.

(a) Bubble sort with early termination. Is the sorting in your case early terminated? Explain the details.

(b) Insertion sort.

- END of Assignment 1 -